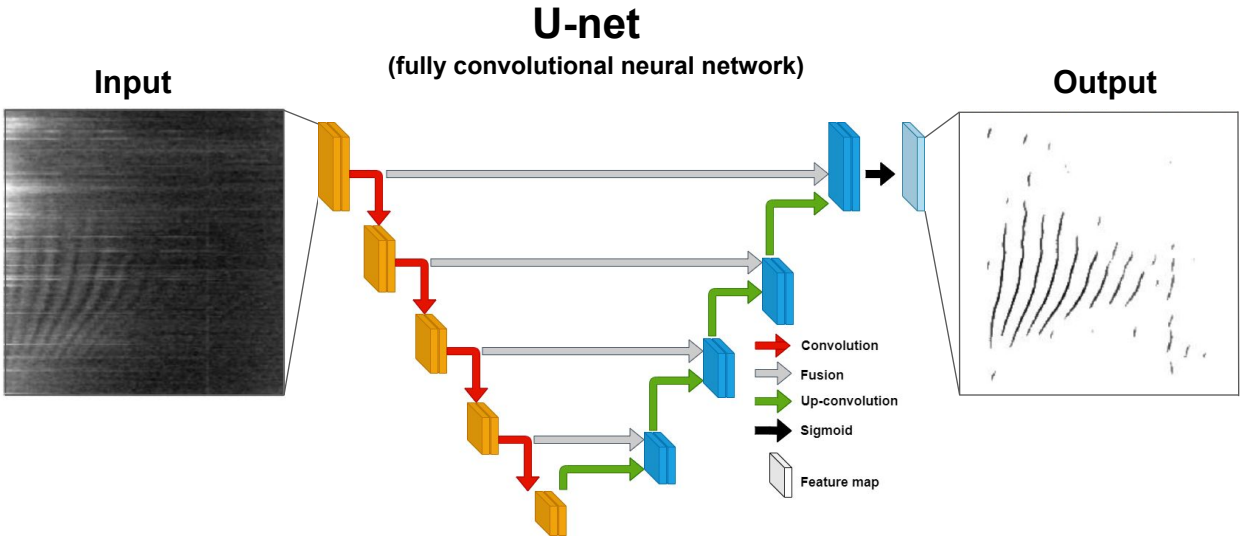
Automatic detection of Ionospheric Alfvén Resonances in magnetic spectrograms using U-net

Graphical Abstract

Automatic detection of Ionospheric Alfvén Resonances in magnetic spectrograms using U-net

Paolo Marangio,Vyron Christodoulou,Rosa Filgueira,Hannah F. Rogers,Ciarán D. Beggan



Highlights

Automatic detection of Ionospheric Alfvén Resonances in magnetic spectrograms using U-net

Paolo Marangio,Vyron Christodoulou,Rosa Filgueira,Hannah F. Rogers,Ciarán D. Beggan

- Application of U-net neural network for image segmentation within the context of geomagnetic data analysis
- Automatic identification of geophysical features in magnetic spectrograms
- U-net provides rapid training and generation of predictions on unseen data

Automatic detection of Ionospheric Alfvén Resonances in magnetic spectrograms using U-net

Paolo Marangio[a], Vyron Christodoulou[b,*], Rosa Filgueira[a], Hannah F. Rogers[c] and Ciarán D. Beggan[b,**]

[a]Edinburgh Parallel Computing Centre, The University of Edinburgh, Bayes Centre, 47 Potterrow, EH8 9BT Edinburgh, United Kingdom
[b]British Geological Survey, The Lyell Centre, Research Avenue South, EH14 4AP Edinburgh, United Kingdom
[c]School of GeoSciences, The University of Edinburgh, Grant Institute, James Hutton Road, EH9 3FE Edinburgh, United Kingdom

ARTICLE INFO

Keywords:
Algorithms
Data Processing
Geophysics
Image Analysis
Parallel and High-Performance Computing

ABSTRACT

Ionospheric Alfvén Resonances (IARs) are weak discrete non-stationary Alfvén waves along magnetic field lines, at periods of ∼0.5–20 Hz, that occur during local night-time, particularly during low geomagnetic activity. They are detectable through time-frequency analysis (spectrograms) of measurements made by sensitive search coil magnetometers. The IARs are generated by the interaction of electromagnetic energy partially trapped in the Earth-ionosphere cavity with the main geomagnetic field and their behavior provides proxy information about atmospheric ion density between 100–1000 km altitude. Limited methods exist to automatically detect and analyse their properties and behavior as they are difficult to extract using standard image and signal processing techniques. We present a new method for the detection of IARs based on the fully convolutional neural network U-net. U-net was chosen as it is able to perform accurate image segmentation and it can be trained in a supervised fashion on a relatively small labeled dataset utilizing data augmentation. We show that the resulting predictive model generated by training the U-net is able to detect IAR signals while mislabelling considerably less noise than other data analysis methods. We achieved our best results by using a training set of 178 hand-digitized examples from high-quality spectrograms measured at the Eskdalemuir Geophysical Observatory (UK). We find that the network converges in ten iterations with a final intersection over union (IoU) metric of 0.9 and a training loss of below 0.2. We use the trained network to extract IARs from over 2300 images, covering six years of search coil magnetometer data measured at the Eskdalemuir Observatory. U-net can also automatically handle missing data or days without IARs, giving a null result as expected. This constitutes the first use of a neural network for pattern recognition of unstructured image data such as spectrograms containing IAR signals, though the method is applicable to other types of resonances or geophysical features in the time-frequency domain.

1. Introduction

The Earth has a large-scale, approximately dipolar, main magnetic field that stretches for thousands of kilometres from its surface into space, passing through the conductive ionosphere out to the magnetosphere. On timescales ranging from months to years, the core field is responsible for driving magnetic field changes, while at periods ranging from seconds to hours, these changes are driven by solar wind interactions with the main field. Between 1 and 100 Hz there are a number of natural resonance phenomena detectable, generated by the reflection/refraction of electromagnetic waves between the conductive surface and the ionosphere. These resonances are known as the Schumann

*Corresponding author
**Principal corresponding author
vyronc@bgs.ac.uk (V. Christodoulou); r.filgueira@epcc.ed.ac.uk (R. Filgueira); h.f.rogers@sms.ed.ac.uk (H.F. Rogers); ciar@bgs.ac.uk (C.D. Beggan)
ORCID(s): 0000-0003-3835-3891 (V. Christodoulou); 0000-0002-5715-3046 (R. Filgueira); 0000-0002-1508-2833 (H.F. Rogers); 0000-0002-2298-0578 (C.D. Beggan)

Resonances (Schumann, 1952) and Ionospheric Alfvén Resonances (IARs) (Polyakov and Rapoport, 1981; Trakhtengerts and Feldstein, 1981; Lysak, 1988).

We focus here on the Ionospheric Alfvén Resonances, which are magnetic field vibrations (i.e. waves) in the range from around 0.5 to 20 Hz (Belyaev et al., 1989). At middle and low latitudes they are produced, indirectly, by the leakage of energy from lightning strikes into near-Earth space (Nosé et al., 2017). At high latitudes, they play a role in the modulation of magnetospheric signals (e.g. Demekhov et al., 2000). They have amplitudes in the picoTesla (pT) range and can be detected on Earth's surface using search coil magnetometers. These instruments are very sensitive to rapid magnetic field variations but cannot be used for long-term or near-DC measurements (i.e. time-varying average of the full magnetic value of Earth's field). Raw data from such instruments are typically processed using a Fast Fourier Transform (FFT) to create one-dimensional periodograms. Multiple periodograms are then stacked into a matrix that can be visualized as a spectrogram. IARs appear as repeating fringes of higher intensity magnetic field strength that change slowly over a few hours (see panel a in Fig. 1) and have been detected at ground stations across the world from low latitudes in Greece (e.g. Bösinger et al., 2002) to Svalbard at high latitudes (Semenova and Yahnin, 2008).

In geophysical terms, the occurrence of IARs and their specific vibration frequencies allow certain properties such as the ion density of the upper atmosphere from 100 to 1000 km to be estimated. This is a region of the atmosphere that is difficult to remotely sense otherwise, making automatic identification and extraction of IAR signals a useful tool for investigating the night-time dynamics of the local atmosphere (e.g. Hebden et al., 2005). Indeed, surprisingly complex and as yet unexplained behaviors have been observed in IARs (e.g. Beggan and Musur, 2018). The key parameters of interest are the frequencies (f) the IARs occur at, and the distance between fringes in frequency (known as Δf), which are both controlled by the density of the ionosphere through which the waves pass (Molchanov et al., 2004).

Within the geomagnetic research community, there are no universal standards for the analysis of IAR signals. Several methods based on signal and image processing techniques have been developed for the semi-automated detection (i.e. labelling) of IARs and computation of parameters of interest (Odzimek et al., 2006; Beggan, 2014). However, not only do these methods require the tuning of several thresholds and parameters, but they are also prone to noise detection regardless of whether IARs are present or not. The aim of this work is therefore to develop an alternative method for the automated detection of IARs based on machine learning.

The work is structured as follows: in Section 2 we introduce the methodology of extracting IARs and provide an overview of the available training data; in Section 3 we give an overview of how to train and test the U-net algorithm and our strategy for confirming the correct behavior of the neural network. In Section 4 we report the results of our experiments, discussing them briefly in Section 5.

2. Extracting IARs from spectrograms

2.1. Using signal and image processing techniques

In September 2012, two search coil magnetometers were installed at the British Geological Survey's Eskdalemuir Observatory, situated in a rural region of the Scottish Borders, UK. These instruments continuously sample changes in the magnetic field of the Earth at 100 times per second (i.e. 100 Hz). The sensors capture geophysical information about IARs although signal processing using an FFT is required to uncover them. To process the raw data, a Butterworth bandpass-filter between frequencies of 0.5 and 10 Hz is applied. After filtering, 100 seconds of data (corresponding to 10,000 samples) are converted into a periodogram using a bespoke Welch periodogram algorithm with a Hanning window applied. For each 24 hour period, 864 periodograms are stacked to produce a spectrogram image. Figure 1 shows examples of spectrograms from the Eskdalemuir Observatory captured during 2012 and 2013. The color indicates the power of the signal in logarithm of pT per square root of Hz, which corresponds to the strength of the magnetic field at a particular frequency and time.

Panel (a) in Figure 1 shows an example of clearly visible IARs (thin bright 'fringes' between 0.5 and 6 Hz), the first Schumann Resonance (broad bright region at 8 Hz) and a magnetospheric pulsation (around 0.5–2 Hz, 13:00 to 17:00 UT). On the other hand, panel (c) shows a day without obvious IARs, while panel (d) illustrates the issue of occasional man-made noise generated by an unknown source pulsing at 1 Hz.

Using the methodology of Beggan (2014), panels (b–d) show a prior attempt to automatically identify and delineate IARs based on signal and image processing techniques alone. This approach relies heavily on peak detection of the IARs as they rise above the general background level and the joining of these peaks using image dilation and erosion algorithms with fixed thresholds based on manual experimentation. The labels generated from this method are overlaid on the original spectrogram, where the black pixels indicate positions in the image that correspond to estimated IAR signals.

As can be observed, while the identification of IARs in panel (b) is adequate, the results in panel (c) are not. In panel (d), we show an example of 1 Hz man-made contamination to illustrate other types of noise that are occasionally present in the dataset. We note that no IARs are present in panel (d), though the algorithm does pick out the 1 Hz harmonics. At the time of their study, Beggan (2014) had around 14 months of magnetic data available, which was used to create statistics of the behavior and occurrence of IARs. However, the method was not wholly successful so further improvements were sought, particularly to remove the reliance on manually set thresholds. The emergence of machine learning (ML) techniques for image segmentation in recent years prompted an investigation of their utility for IAR extraction.

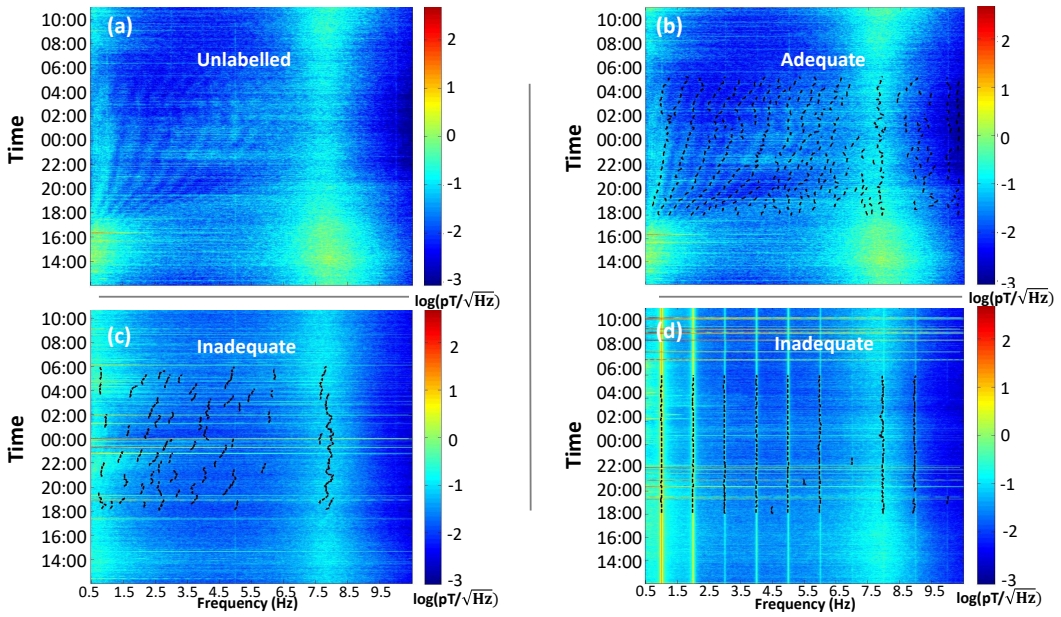


Figure 1: Performance of original data analysis method from Beggan (2014) on three sample days. (Panel a). 14-15 February 2013: Unlabeled spectrogram image showing IARs as alternating darker and brighter blue patterns occurring between 18:00 and 06:00. Note also the first Schumann Resonance at ~ 8 Hz. (Panel b) As panel (a) but labelled (black dotted lines) with IARs. (Panel c) 13–14 June 2013: Example of poor performance of IAR labelling. (Panel d) 25-25 May 2013: Labelling of vertical lines associated with local man-made electrical interference.

2.2. Using Machine Learning techniques for image segmentation

An alternative to the approach based on signal and image processing techniques proposed by Beggan (2014) can be found in machine learning, which is a field of study concerned with the automation of learning using mathematics and statistics. In particular, we seek a ML tool that is capable of robustly identifying the 'fringe' pattern of IARs in spectrograms. Machine learning has recently been a driving force behind the huge progress made in tackling a variety of computer vision problems, such as object detection, motion tracking, action recognition, human pose estimation and semantic segmentation (see Voulodimos et al. (2018) for a review). The results of these applications have been so promising that the whole field of computer vision is shifting towards being ML-based, relinquishing the need for pipelines of specialized, hand-crafted methods.

Deep learning is a branch of ML that is concerned with the formulation of computational models that include multiple, successive processing layers, which allows data to be represented using an abstraction hierarchy; the typical example for this is a neural network. Modern neural networks tend to fall into one of the following categories: convolutional neural networks, recurrent neural networks, recursive neural networks and generative adversarial networks. The application of neural networks to geological problems has increased in recent years: Conway et al. (2019) used a network to invert magnetotelluric data to determine subsurface conductivity, Karimpouli and Tahmasebi (2019)

segmented different mineral phases in thin-section images and Miller et al. (2018) identified cirrus clouds in satellite imagery.

Convolutional neural networks (CNNs) constitute a specialized type of neural network that has been successfully employed in a wide range of ML tasks, including classification of text and images. These networks are inspired by the structure of the biological visual system (Hubel and Wiesel, 1962; Fukushima, 1980) and it is therefore unsurprising that CNNs constitute one of the most important types of ML models for visual understanding. CNNs have been used to identify micro-seismic events (Wilkins et al., 2020) and to classify rock type in drilled cores (Baraboshkin et al., 2020), as well as to detect volcanic craters on Mars (Palafox et al., 2017). In particular, a subcategory of CNNs known as fully convolutional networks (FCNs) has demonstrated excellent performance for the semantic segmentation task (Long et al., 2015).

U-net is a FCN originally described by Ronneberger et al. (2015) and winner of the Cell Tracking Challenge at the International Symposium on Biomedical Imaging in 2015. This FCN is able to perform classification at the pixel level while maintaining global structure within an image. In short, it does this by first downsampling the input image and learning its high-level features, followed by upsampling and localization of the identified image features. Moreover, U-net can leverage data augmentation in order to compensate for cases where only small labelled datasets are available, such as in the biomedical imaging domain. With a curated dataset of labelled images it is possible to efficiently train U-net in a supervised fashion.

3. Training and testing U-net

Since their installation, the search coil magnetometers at the Eskdalemuir Observatory have recorded data for ∼95% of the time, allowing spectrograms for 2312 calendar days between 01-Sep-2012 and 01-Jan-2019 to be generated. A single spectrogram consists of 864 periodograms with a time resolution of 100 seconds and a frequency resolution of ∼0.02 Hz per point, using the Nyquist frequency of 50 Hz and a 4096-point FFT. The spectrograms are saved as images with a fixed size of 1100 x 1400 pixels.

We manually selected 178 spectrograms with well-defined IARs to form our initial dataset, based on visual inspection of the 2312 images. Consistent with the use of U-net by Ronneberger et al. (2015), we used grayscale images instead of color ones in order to simplify the processing of the input images during training. We truncated the spectrograms between 18:00 and 06:00 Universal Time (UT) as IARs do not occur during daylight hours (though we ignore seasonal changes at present) and also to reduce the input image size to 1100 x 700 pixels.

Two sets of labelled images were created from the 178 manually selected images. The first set of labelled images (Training Set A) were based on the outputs of the Beggan (2014) method, while the second set (Training Set B) were manually drawn on top of the training images by eye using a graphics package. The labelled images consist of either

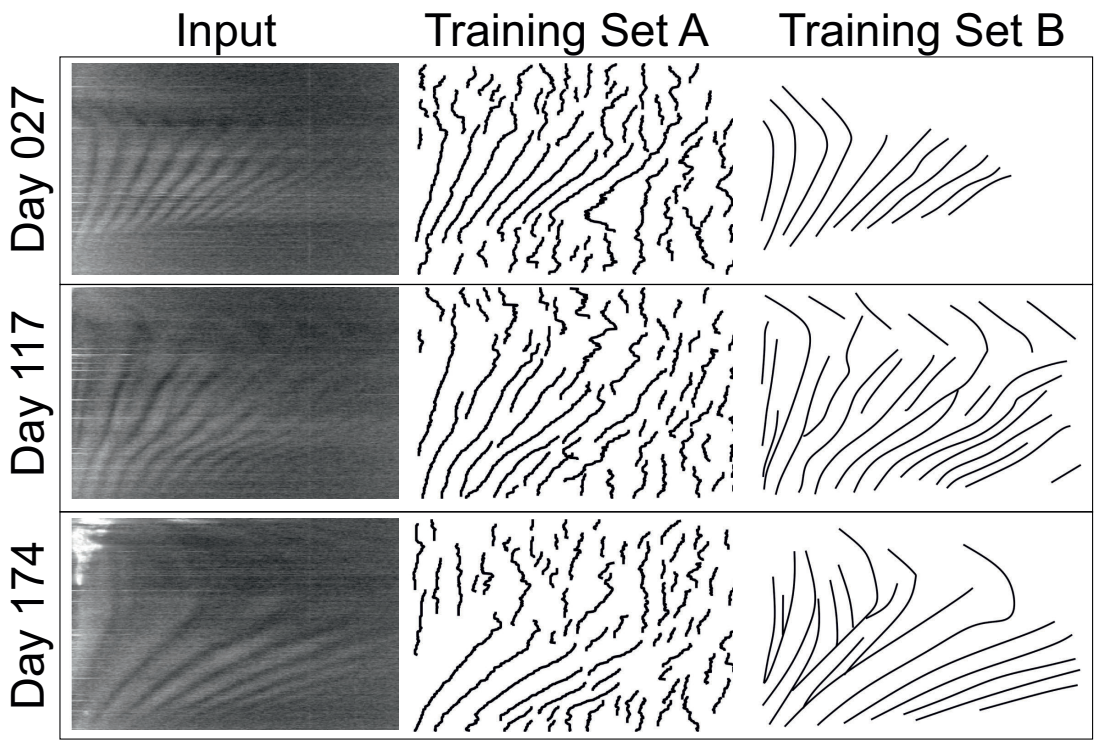


Figure 2: Example of three (out of 178) training and labelled images. (Left column). Training images with IARs. (Centre column). Training Set A labelled images based on the results from Beggan (2014). (Right column). Training Set B labelled images created from visual interpretation of the IARs. Day 027: 30-Jul-2013; Day 117: 19-Jul-2016; Day 174: 04-Sep-2018.

black or white pixels, where black pixels correspond to IARs and white pixels correspond to background (i.e. no signal) in the associated training example. Training Set A tends to capture the numerical peaks while Training Set B is based on visual interpretation by the scientist responsible. Both training sets, in effect, highlight the position of the fringes, which are generally brighter than the background. Figure 2 shows three example days with the unlabeled training image in the left column, the IARs detected using the method by Beggan (2014) in the central column (Training Set A) and the manually picked IARs in the right column (Training Set B). The examples demonstrate that even with reasonably clear IARs, the definition of their location and extent is actually quite subjective.

We coded up the U-net FCN using Keras and TensorFlow in Python 3.6 using freely available packages. The code was run under various Linux environments ranging from laptops to multicore virtual machines and on the University of Edinburgh Cirrus HPC platform, which included facilities for GPU processing (EPCC, 2019).

3.1. U-net hyperparameter selection, metrics and model evaluation strategy

In order to efficiently train a neural network like U-net using an optimization algorithm with backpropagation (Rumelhart et al., 1986), a number of so-called ‘hyperparameters‘ must be selected. Without sufficient tuning, the

optimization algorithm may not converge to a good solution, may converge slowly or may not even converge at all. Furthermore, another key aspect of training a neural network lies in the choice of model evaluation metrics and strategy. Evaluation metrics help determine how the model generated by training the network generalizes on unseen data.

We based our study on a publicly available implementation of U-net devised by Zhixu (2017) who segmented neuronal structures in electron microscopic stacks as part of the (now public) dataset presented at the International Symposium on Biomedical Imaging in 2015 (Table 1). For our training of U-net on the labelled IAR datasets, a series of experiments were made to search for optimal hyperparameter values, as detailed in this section.

Table 1
List of initial hyperparameters values for training U-net based on the implementation of the neural network by Zhixu (2017).

Hyperparameter	Value
Batch size	2
Dropout	0.5
# Epochs	5
Learning rate	10^{-4}
Loss function	Binary cross-entropy
Optimizer	Adam
Weight initializer	he normal

Although accuracy is the most common metric for monitoring the training of a ML algorithm and its performance on test data, it is less useful and informative in situations where there is a class imbalance; that is, when there are vastly different numbers of pixel samples per image between the different target classes (the background and IAR signal classes in our case). Another example is if there is a substantial variation in the cost of different prediction errors (Chawla, 2005; Hossin and Sulaiman, 2015). Accuracy is clearly not suitable for our application, as a single labelled image typically contains about 10 times more white pixels (background class) than black pixels (IAR class). Hence, using accuracy would give a larger influence to the background class compared to the IAR class. In this scenario it would be easy to achieve a high accuracy without correctly segmenting the IAR i.e. by naively labelling every pixel as background rather than IAR.

Instead, the Intersection over Union (IoU) metric is used to quantify the percent overlap between the ground truth (i.e. training labels) provided before training and the prediction output generated during training (Levandowsky and Winter, 1971). A true positive (TP) represents a pixel that is correctly predicted to belong to the given class according to the ground truth, whereas a true negative (TN) represents a pixel that is correctly identified as not belonging to the given class. Moreover, a false positive (FP) represents a pixel that is incorrectly predicted to belong to the given class, while a false negative (FN) represents a pixel that is incorrectly predicted as not belonging to the given class. The IoU metric is calculated by counting the number of pixels that are correctly predicted (i.e. pixels with the same location and class label in both the ground truth and prediction output) divided by the sum of the number of pixels present across

both the ground truth and prediction output: IoU = TP / (TP + FP + FN).

We chose k-fold cross-validation as the method used to estimate model performance. K-fold cross-validation is a resampling method where a model of interest is repeatedly re-fit to different selections of samples from the training set in order to obtain additional information about the fitted model (James et al., 2013). In particular, the dataset is split into k parts and the training process is repeated k number of times. For each of these k iterations, a different 'fold' of the dataset is picked to be the validation set and the remaining k-1 folds are used as training set. On each of these iterations, the value for the evaluation metric and loss function, including the loss on the validation set, are computed. Ideally, the average of these values is the same as that obtained with the model trained on the entire dataset directly without k-fold cross-validation.

Empirically, a value of 5 or 10 for k is known to generate estimates for the loss function and evaluation metric that do not exhibit high bias or variance (James et al., 2013). Based on our experiments conducted with k set to 2, 5 or 10, it was found that a value of 2 exhibits the best trade-off between training dynamics (i.e. optimal values for training loss and IoU evaluation metric) and amount of variation between folds on the training set. Hence, all k-fold cross-validation tests were conducted with a value of k set to 2. It should be noted that while choosing a value of k is critical for model estimation (i.e. to provide a statistically-sound framework for choosing the best values for the hyperparameters), the final predictive model is generated by training U-net on the entire dataset.

3.2. Tuning the number of epochs

In the context of this work, an epoch constitutes a single forward and backward pass through the entire training set by the U-net algorithm. It is generally the case that increasing the number of epochs leads to improvements in the values of the evaluation metric, regardless of whether values of other hyperparameters are themselves optimal. Hence, as a first step the number of epochs was tuned while keeping the other hyperparameter values fixed. The U-net was independently trained for 1, 3, 5 and 10 epochs. The resulting training loss and IoU values are shown in Table 2.

By considering the final loss function and IoU values reached at the end of training along with a qualitative assessment of the predictions on test data (data not shown), it appears that the neural network already performs reasonably well by just setting the number of epochs to 3. Further increasing the number of epochs does not have a significant impact on the training loss, the IoU evaluation metric or the quality of predictions on test data, though increasing the number of epochs brings a marginal improvement. We decided to set the number of epochs to 10 for subsequent experiments, as this provides certainty that the best possible final training values are always achieved regardless of the initial values chosen for the hyperparameters. In addition, to test whether the predictive model had overfitted to the training data and, as a result, failed to generalize well on unseen data, U-net was separately trained for 100 epochs. We observed that validation loss plateaued after 50 epochs, indicating that this is the point at which the neural network is

Table 2
Training loss and IoU values from U-net trained on IARs Training Sets A and B with different number of epochs. U-net implementation was independently trained for 1, 3, 5 or 10 epochs. Final training loss and IoU values recorded at the end of training are reported.

	Number of epochs			
	1	3	5	10
A:Training loss	0.3800	0.2910	0.2862	0.2767
A:Training IoU	0.7719	0.8193	0.8217	0.8270
B:Training loss	0.2503	0.2026	0.1960	0.1874
B:Training IoU	0.8735	0.8953	0.8970	0.9008

Table 3
List of hyperparameter values to be explored with grid search while training U-net on IARs Training Set A with k-fold cross-validation. Note, *u.* stands for uniform, while *n.* stands for normal. A value of 1.0 for dropout means that no dropout is applied (Srivastava et al., 2014).

Hyperparameter	Values to be explored
Batch size	2, 4, 8, 16, 32
Dropout	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0
Learning rate	10^{-2}, 10^{-3}, 10^{-4}
Optimizer	SGD with momentum, RMSprop, Adam
Weight initializer	lecun u., glorot n., glorot u., he n., lecun n., he u. and orthogonal

starting to overfit to the training data. This suggests that U-net is certainly not overfitting to either of the IARs datasets when trained for 10 epochs in subsequent experiments.

3.3. Further U-net hyperparameter tuning

The remaining hyperparameters needed for training U-net, namely batch size, dropout, optimizer, learning rate and weight initializer were tuned using a grid search (Bergstra et al., 2011) in combination with k-fold cross-validation. The first step of any grid search experiment is the definition of the values of the hyperparameters to be explored (Table 3). In particular, we decided to compare the performance of three of the most popular optimization algorithms, namely stochastic gradient descent (SGD) with momentum (Qian, 1999), RMSprop (Hinton et al., 2014) and Adam (Kingma and Ba, 2017). We also tested several popular weight initializers that are available through the Keras neural network library (Chollet et al., 2015). For the learning rate, we decided to explore values falling in the range ($10^{-6} <$ learning_rate < 1.0), which includes values that are known to work well for neural networks with standardized inputs (Bengio, 2012). Finally, batch size values were explored as multiples of 2 (Patterson and Gibson, 2017) up to 32, since memory allocation issues ensued when a value of 64 was used in our particular GPU configuration.

Based on a preliminary test, training U-net with a single permutation of the 5 hyperparameters with k set to 2 and

Table 4
Results of grid search for tuning batch size, optimizer and learning rate while training U-net on IARs Training Set A with k-fold cross-validation.

	Permutation		
Hyperparameter	1st	2nd	3rd
Batch size	2	2	2
Learning rate	10^{-4}	10^{-4}	10^{-2}
Optimizer	Adam	RMSprop	SGD with momentum
Cross-validation score	0.2891	0.2926	0.2927

Table 5
Results of grid search for tuning dropout and weight initializer while training U-net on IARs Training Set A with k-fold cross-validation.

	Permutation		
Hyperparameter	1st	2nd	3rd
Dropout	1.0	0.5	1.0
Weight initializer	he uniform	he uniform	lecun normal
Cross-validation score	0.2889	0.2918	0.2930

number of epochs set to 10 took ∼100 seconds on a single GPU (i.e. NVIDIA Tesla V100-PCIE-16GB). Considering that the total number of permutations of the hyperparameter values is 3150 (5 values for batch size, 10 values for dropout, 3 values for learning rate, 3 values for optimizer and 7 values for weight initializer), such an experiment would take ∼88 hours. Due to limitations placed by the computer facility on maximum allocatable computing time, the immediate solution was to split the grid search into two non-exhaustive searches with permutations based on different subsets of the hyperparameter values.

A grid search experiment using k-fold cross-validation was performed, with the initial values for dropout and weight initializer of the U-net implementation proposed by Zhixu (2017), while changing values for batch size, optimizer and learning rate. The 3 permutations of hyperparameter values with the best cross-validation scores are listed in Table 4. The cross-validation score is calculated as the average of validation loss over the two folds (i.e. since k is set to 2) for a given permutation of the hyperparameters. An additional, non-exhaustive grid search experiment was then performed while changing values for dropout and weight initializer on top of the best permutation of values for batch size, learning rate and optimizer identified in the previous grid search. The three permutations of hyperparameter values with the best cross-validation scores identified through the second grid search are listed in Table 5. By comparing Tables 4 and 5, it can be seen that the best permutations from the two individual grid searches have very similar cross-validation scores. It must be noted that as these experiments both technically constitute non-exhaustive grid searches, it is still not possible to definitely argue that the hyperparameter values identified in the two grid searches are indeed optimal. However, the

fact that the best permutations from both grid searches have similar cross-validation scores, combined with the fact that the optimal values identified by tuning 3 out of 5 hyperparameters in the first grid search correspond to the initial hyperparameter values used in this work suggests that the choice of initial hyperparameter values is reasonably good.

4. Results

U-net was separately trained using the two different labelled IAR datasets (Training Set A and Set B). The final training loss and IoU values after 10 epochs were 0.2767 and 0.8270 for Training set A, and 0.1874 and 0.9008 for Training Set B (see Table 2). It is important to recall that the 'ground truth' images used for training U-net on the IARs Training Set A are generated using imperfect labels, as the definition of IARs is probabilistic rather than discrete. Therefore, the U-net output is expected to contain predicted noise or signals that are not associated with IARs. However, unlike the method of Beggan (2014), the U-net classifier assigns a probability value to each of the pixels in the prediction output. This means that it is possible to remove or reduce some of the 'noise' from the prediction output by setting all the values that are greater than a given (inverse) probability or threshold to 1 (i.e. a white pixel, representing a pixel that has been classified as background).

Different thresholds in the range 0.4-0.9 were tested on the outputs of Training Sets A and B. By visually comparing the thresholded prediction output against the ground truth image, we determined that thresholds of 0.5 and 0.8 offer the best trade-off between correct signal detection and noise removal for the trained U-net based on Training Set A and B, respectively. Figure 3 shows the outputs from the network trained for Training Set A and Set B. The left-hand panels show the same grayscale test image, with the output from each trained U-net in the second column. The third and fourth column show the thresholded versions of the output image. The upper row (from Training Set A) is thresholded at 0.5 and 0.6, respectively, while the lower row (from Training Set B) has threshold values of 0.8 and 0.9 applied to the respective output images. The results are surprisingly similar, given the large visual differences between the labelled training datasets.

In order to confirm that the U-net has actually been trained to detect meaningful patterns in new unseen data, we also tested a spectrogram with no IARs visible as input. Figure 4 shows an example of the output generated when such an image is presented to the U-net algorithm after being trained on Training Set A. As required, the prediction on the negative control test image contains low-value gray pixels that are strongly blurred, suggesting that for such pixels the neural network is unable to classify IARs with sufficient certainty. By applying a threshold of 0.4 or 0.5, it is possible to nearly obtain an almost empty output. This is the outcome we desire for an image that contains no IARs. This test provides further support for choosing a threshold to apply to the predicted output generated from the U-net. We point out that further processing steps are required to extract the f and Δf parameters of interest from coherent IAR signals, so small amounts of spurious signal (as in images under 'Thresholded Output' in Figure 3) will be removed at a later

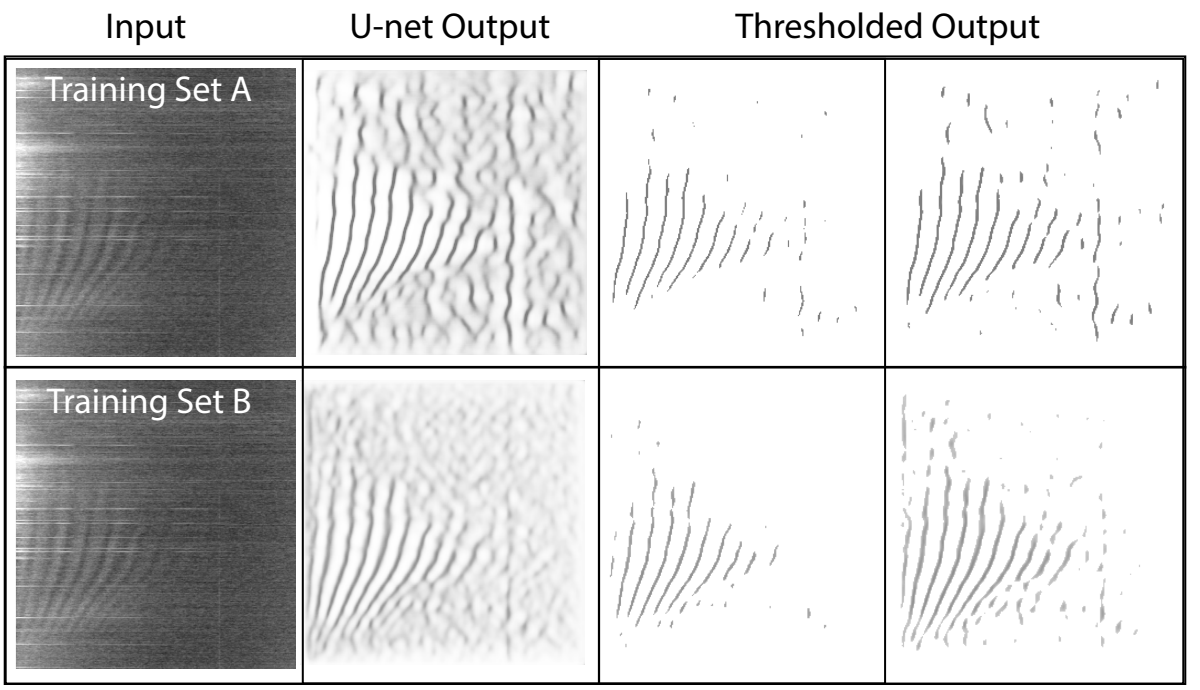


Figure 3: The effect of different thresholds applied to the U-net output (for an unseen test image: 15-Sep-2012). Top row is for Training Set A and bottom row is for Training Set B. The U-net algorithm was separately trained on each training set for 10 epochs. (Left column). Input test image. (Second column). U-net output. (Third column). Thresholded U-net outputs for 0.5 (upper panel) and 0.8 (lower panel). (Fourth column). Thresholded U-net outputs for 0.6 (upper panel) and 0.9 (lower panel).

stage through additional processing steps to produce the geophysically relevant parameters.

Using the trained networks we generated predictions for the remaining ∼2100 spectrograms from the full IARs dataset (i.e. those not used for training).

5. Discussion

During the training of the U-net, we apply a quantitative metric (IoU) and a loss function as the optimization methods by which the neural network is able to find the weights that best map the inputs to the outputs provided during training. More generally, we can also assess the performance of the trained predictive model against the original data analysis method from a qualitative point of view. We can, for example, overlay the thresholded prediction output on test data on top of the 'ground truth' image, as illustrated in Figure 5.

The left hand panels show two example 'Input' spectrograms (i.e. unlabelled images), with the associated labelled test images (i.e. ground truth) shown in the 'Labelled' column. The 'Thresholded' column is the U-net output thresholded with 0.5 (for Training Set A) or 0.8 (for Training Set B). The images in the 'Overlay' column are the result of the superposition of the ground truth image (i.e. 'Labelled' column) over the respective thresholded prediction output (i.e.

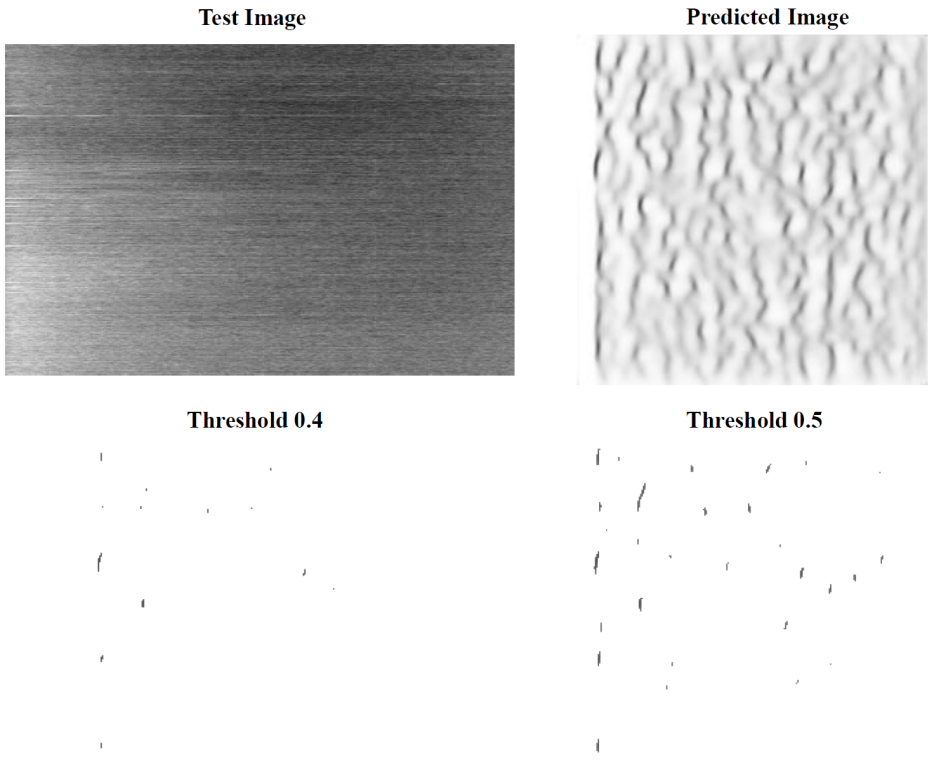


Figure 4: Negative control test with noisy example containing no signal. (Top left panel). Test image (26-Mar-2016) with no visible IARs. (Top right panel). Resulting prediction output from the predictive model generated by training the U-net algorithm with Training Set A. (Bottom left and right panels). Thresholded versions of prediction output using values of 0.4 and 0.5, respectively.

'Thresholded' column) for a given test image. The pixels in such images fall into one of the following three categories: red pixels indicate pixels that are labeled as background in both the ground truth image and the thresholded prediction output (namely TN); black pixels indicate pixels that are predicted as IARs in both images (namely TP); white pixels indicate pixels that are labelled as IARs in the ground truth, but as background in the thresholded prediction output (namely FN).

The white pixels in any given image under the 'Overlay' column essentially highlight a portion of the putative IARs that are present in the labelled images, but absent when the prediction is made on the same test image with the U-net model. This observation highlights the main issues with the labelling methods, namely the presence of spurious noise from the peak detection-based method and the inherent subjectivity associated with the manual digitization-based method, which lead to the erroneous labelling of IAR signals. In the Beggan (2014) (or Training Set A) method, the labelled images have many relatively short or spurious fragmented lines associated with noise (panel c in Fig. 1). In particular, as the white pixels in any given image of the 'Overlay' column for Training Set A tend to be associated with such fragmented lines, the trained predictive model appears to be less prone to predicting noise than the Beggan (2014)

method. In the manually-derived (or Training Set B) method, the labelling (see 'Labelled' column) can be incomplete or extended by the 'eye-of-faith' when humans tend to see patterns where none exist.

It is therefore interesting to find that both U-net models produce very similar fringe patterns (see images in the 'Thresholded' column) despite being based on completely different training sets. It could be argued that Training Set A is very comprehensive and captures all the bright fringes but encapsulates extraneous signal that corresponds to noise, while Training Set B is cleaner but not as extensive, as it has a lower overall number of training pixel samples corresponding to IARs. The general explanation for this result is the iterative nature of machine learning means that chosen patterns are reinforced when they are repeatedly linked to features in the input images; though this often not visually intuitive.

In the case of Training Set A (e.g. in Figure 2), as there are many small features labelled as IAR compared to Training Set B, it seems counter-intuitive that they should both produce similar results. However, the power of machine learning lies in its ability to pick out a desired signal from noise. When the U-net is trained, the smaller and shorter-lived features tend to be down-weighted in the network's overall response as they do not always correspond to obvious feature in the input training images. We have examined individual node responses within the U-net for training set A and observed the manner in which the 'bright' regions of the training set data are up-weighted. Where there are regions that are labelled but do not correspond to 'bright' patterns, these become down-weighted and assigned a lower probability.

We also note that, as the U-net outputs a probability of a pixel being classed as an IAR feature, stronger probability features can be thresholded by identifying a suitable value. In Figure 3, the 'U-net Output' column for Training Set A contains 'blurred' (light gray) features as well as well-delineated (dark gray) features. The blurred features thus have a low probability of being IARs. The 'Thresholded Output' columns show the result of removing these by simply ignoring features with low probability.

With Training Set B, that U-net has learned to associate a label with a fringe only when a fringe or bright region is present in the interrogated image. In Figure 3, after thresholding, this U-net labels fewer pixels in the right side of the output image (corresponding to the high frequency values) compared to the output image for training set A.

That the U-net produces such robust results on relatively few images is also remarkable. Indeed, the optimal number of examples required to robustly train a deep neural network is a matter of ongoing research, though it appears possible to be successful with relatively small datasets ranging from hundreds to thousands, rather than millions, of training examples (Chen et al., 2017; Pesce et al., 2019). Considering this, it could be argued that with the small datasets used here (178 training examples), a small increase in dataset size could still have a noticeable impact on performance.

For this application it can be concluded that, from a qualitative point of view, the trained predictive model (using either Training Set A or B) exhibits an ability to identify and segment most of the IAR signals, with a higher signal to

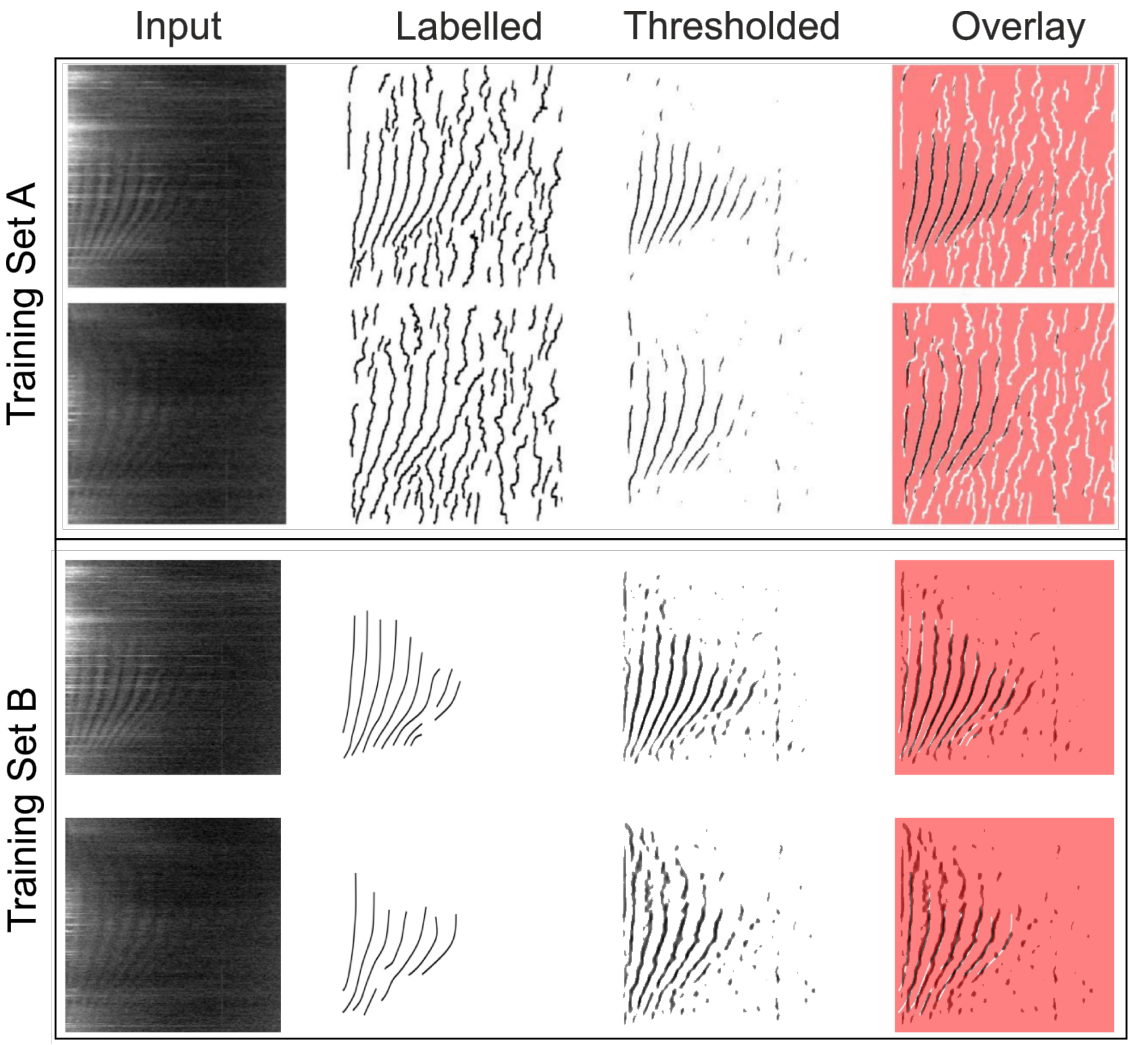


Figure 5: Qualitative assessment of performance of original data analysis method against predictive model generated by training U-net on either Training Set A (top) or B (bottom). (First column from left). Spectrograms for 05-Sep-2012 and 06-Sep-2012 drawn from the test set. (Second column). Respective 'ground truth' images. (Third column). Respective thresholded (with value of 0.5 or 0.8 for Training Set A or B, respectively) prediction output. (Fourth column). Overlay of the ground truth image in the second column on the thresholded prediction output in the third column. Figure was generated using code adapted from Zak (2019).

noise ratio than the method presented in Beggan (2014), and is able to reject most unwanted or spurious noise. This makes U-net a useful method for automatically extracting IARs. We also suggest that other similar types of geophysical signals visualized as spectrograms could be extracted using U-net, for example recurring tremors in seismic data or acoustic emissions from rock fracturing experiments.

6. Conclusion

In this work, a novel application of the U-net neural network for automatic image segmentation of magnetic search coil data has been described. The objective of the work was to identify IARs and to efficiently train a neural network on a small, curated dataset in order to improve on an existing data analysis method. One challenge was to choose the best set of hyperparameter values that would enable the neural network to create a reasonably good mapping from input to output image. This was achieved using a robust model evaluation strategy, namely k-fold cross-validation, in combination with a grid search experiment. The second challenge was to make the network robust in the presence of faint or no IARs.

Two predictive models were generated by separately training the U-net algorithm on two different datasets. The first dataset was derived from an automated process based on peak detection (Beggan, 2014), while the second was based on visually identified IARs drawn by hand. Both trained U-net models produced surprisingly similar results and perform better than the original data analysis method. The networks are also robust to noise or missing data.

Moreover, the neural network is fast: it takes as little as ∼132 seconds on average to train U-net for 10 epochs with a dataset of 178 training examples on an Nvidia Tesla V-100 GPU. The resulting predictive model only takes ∼58 milliseconds to generate a prediction output for a test image. The approach and methods that have been used in this work on U-net for the magnetic spectrograms are also applicable to the automatic detection and identification of other geophysical features with distinctive spectral or visual patterns.

In this type of application, the use of good quality labelled data in the training phase allows expert-elucidated domain knowledge to be implicitly imparted to the neural network. Although tuning of hyperparameters should always be performed, as it allows an optimal fit to be determined, the quality of the training data is vital for a successful outcome. Based on the results generated in this study, we expect U-net to be of value for the analysis of geophysical datasets that require segmentation of a desired signal, assuming they have clearly definable patterns with minimal noise.

7. Computer Code and Data Availability

The software and example datasets for implementing U-net on the IARs spectrograms can be accessed at the following repository: `https://github.com/marangiop/unet`.

CRediT authorship contribution statement

Paolo Marangio: Investigation, Methodology, Software, Writing - Original draft, Writing - Review & Editing. **Vyron Christodoulou:** Methodology, Supervision, Investigation, Software, Writing - Review & Editing. **Rosa**

Filgueira: Conceptualization, Supervision, Resources, Writing - Review & Editing. **Hannah F. Rogers:** Data Curation, Validation, Writing - Review. **Ciarán D. Beggan:** Conceptualization, Resources, Data Curation, Validation, Writing - Review & Editing.

Acknowledgements

We wish to thank the reviewers and the editors for their constructive comments which improved the final manuscript.

References

1. Baraboshkin, E.E., Ismailova, L.S., Orlov, D.M., Zhukovskaya, E.A., Kalmykov, G.A., Khotylev, O.V., Baraboshkin, E.Y., Koroteev, D.A., 2020. Deep convolutions for in-depth automated rock typing. Computers& Geosciences 135, 104330. doi:10.1016/j.cageo.2019.104330.
2. Beggan, C.D., 2014. Automatic detection of ionospheric Alfvén resonances using signal and image processing techniques. Annales Geophysicae 32, 951–958. doi:10.5194/angeo-32-951-2014.
3. Beggan, C.D., Musur, M., 2018. Observation of Ionospheric Alfvén Resonances at 1-30 Hz and Their Superposition With the Schumann Resonances. Journal of Geophysical Research: Space Physics 123, 4202–4214. doi:10.1029/2018JA025264.
4. Belyaev, P.P., Polyakov, C.V., Rapoport, V.O., Trakhtengerts, V.Y., 1989. Experimental studies of the spectral resonance structure of the atmospheric electromagnetic noise background within the range of short-period geomagnetic pulsations. Radiophysics and Quantum Electronics 32, 491–501. doi:10.1007/BF01058169.
5. Bengio, Y., 2012. Practical recommendations for gradient-based training of deep architectures. arXiv:1206.5533 [cs]. URL: https://arxiv.org/pdf/1206.5533.pdf.
6. Bergstra, J.S., Bardenet, R., Bengio, Y., Kégl, B., 2011. Algorithms for Hyper-Parameter Optimization, in: Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems 24. Curran Associates, Inc., pp. 2546–2554. URL: https://papers.nips.cc/paper/4443-algorithms-for-hyper-parameter-optimization.
7. Bösinger, T., Haldoupis, C., Belyaev, P., Yakunin, M., Semenova, N., Demekhov, A., Angelopoulos, V., 2002. Spectral properties of the ionospheric Alfvén resonator observed at a low-latitude station (L= 1.3). Journal of Geophysical Research 107, 1281. doi:10.1029/2001JA005076.
8. Chawla, N.V., 2005. Data Mining for Imbalanced Datasets: An Overview, in: Maimon, O., Rokach, L. (Eds.), Data Mining and Knowledge Discovery Handbook. Springer US, Boston, MA, pp. 853–867. doi:10.1007/0-387-25465-X_40.
9. Chen, S.W., Shivakumar, S.S., Dcunha, S., Das, J., Okon, E., Qu, C., Taylor, C.J., Kumar, V., 2017. Counting apples and oranges with deep learning: A data-driven approach. IEEE Robotics and Automation Letters 2, 781–788. doi:10.1109/LRA.2017.2651944.
10. Chollet, F., et al., 2015. Keras. URL: https://github.com/fchollet/keras.
11. Conway, D., Alexander, B., King, M., Heinson, G., Kee, Y., 2019. Inverting magnetotelluric responses in a three-dimensional earth using fast forward approximations based on artificial neural networks. Computers & Geosciences 127, 44–52. doi:10.1016/j.cageo.2019.03.002.
12. Demekhov, A.G., Trakhtengerts, V.Y., Bösinger, T., 2000. Pc 1 waves and ionospheric Alfvén resonator: Generation or filtration? Geophysical Research Letters 27, 3805–3808. doi:10.1029/2000gl000126.
13. EPCC, 2019. Cirrus Hardware. URL: https://www.cirrus.ac.uk/about/hardware.html.

14. Fukushima, K., 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36, 193–202. doi:10.1007/BF00344251.

15. Hebden, S.R., Robinson, T.R., Wright, D.M., Yeoman, T., Raita, T., Bösinger, T., 2005. A quantitative analysis of the diurnal evolution of ionospheric Alfvén resonator magnetic resonance features and calculation of changing IAR parameters. Annales Geophysicae 23, 1711–1721. doi:10.5194/angeo-23-1711-2005.

16. Hinton, G., Srivastava, N., Swesky, K., 2014. CSC321 - Introduction to Neural Networks for Machine Learning. URL: https://www.cs.toronto.edu/~hinton/coursera/lecture6/lec6.pdf.

17. Hossin, M., Sulaiman, M.N., 2015. A Review on Evaluation Metrics for Data Classification Evaluations. International Journal of Data Mining & Knowledge Management Process 5, 01–11. doi:10.5121/ijdkp.2015.5201.

18. Hubel, D.H., Wiesel, T.N., 1962. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology 160, 106–154.2. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1359523/.

19. James, G., Witten, D., Hastie, T., Tibshirani, R., 2013. An Introduction to Statistical Learning: with Applications in R. Springer Texts in Statistics, Springer-Verlag, New York. doi:10.1007/978-1-4614-7138-7.

20. Karimpouli, S., Tahmasebi, P., 2019. Segmentation of digital rock images using deep convolutional autoencoder networks. Computers & Geosciences 126, 142–150. doi:10.1016/j.cageo.2019.02.003.

21. Kingma, D.P., Ba, J., 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980 [cs] URL: http://arxiv.org/abs/1412.6980.

22. Levandowsky, M., Winter, D., 1971. Distance between Sets. Nature 234, 34–35. doi:10.1038/234034a0.

23. Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440. doi:https://doi.org/10.1109/TPAMI.2016.2572683.

24. Lysak, R.L., 1988. Theory of auroral zone PiB pulsation spectra. Journal of Geophysical Research 93, 5942. doi:10.1029/ja093ia06p05942.

25. Miller, J., Nair, U., Ramachandran, R., Maskey, M., 2018. Detection of transverse cirrus bands in satellite imagery using deep learning. Computers & Geosciences 118, 79–85. doi:10.1016/j.cageo.2018.05.012.

26. Molchanov, O., Schekotov, A., Fedorov, E., Hayakawa, M., 2004. Ionospheric Alfven resonance at middle latitudes: results of observations at Kamchatka. Physics and Chemistry of the Earth, Parts A/B/C 29, 649 – 655. doi:10.1016/j.pce.2003.09.022.

27. Nosé, M., Uyeshima, M., Kawai, J., Hase, H., 2017. Ionospheric Alfvén resonator observed at low-latitude ground station, Muroto. Journal of Geophysical Research: Space Physics 122, 7240–7255. doi:10.1002/2017JA024204. 2017JA024204.

28. Odzimek, A., Kułak, A., Michalec, A., Kubisz, J., 2006. An automatic method to determine the frequency scale of the ionospheric Alfvén resonator using data from Hylaty station, Poland. Annales Geophysicae 24, 2151–2158. doi:10.5194/angeo-24-2151-2006.

29. Palafox, L.F., Hamilton, C.W., Scheidt, S.P., Alvarez, A.M., 2017. Automated detection of geological landforms on Mars using Convolutional Neural Networks. Computers & Geosciences 101, 48–56. doi:10.1016/j.cageo.2016.12.015.

30. Patterson, J., Gibson, A., 2017. Deep Learning: A Practitioner's Approach. 1st ed., O'Reilly Media, Inc.

31. Pesce, E., Withey, S.J., Ypsilantis, P.P., Bakewell, R., Goh, V., Montana, G., 2019. Learning to detect chest radiographs containing pulmonary lesions using visual attention networks. Medical Image Analysis 53, 26 – 38. doi:10.1016/j.media.2018.12.007.

32. Polyakov, S., Rapoport, V., 1981. The Ionospheric Alfvén resonator. Geomagnetism and Aeronomy 21, 816–822. doi:10.1016/0021-9169(90)90010-K.

33. Qian, N., 1999. On the momentum term in gradient descent learning algorithms. Neural Networks 12, 145 – 151. doi:10.1016/

S0893-6080(98)00116-6.

34. Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. arXiv:1505.04597 [cs] URL: http://arxiv.org/abs/1505.04597.

35. Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323, 533. doi:10.1038/323533a0.

36. Schumann, W.O., 1952. Über die strahlungslosen Eigenschwingungen einer leitenden Kugel, die von einer Luftschicht und einer Ionosphärenhülle umgeben ist. Zeitschrift für Naturforschung A 7, 149–154. doi:10.1515/zna-1952-0202.

37. Semenova, N.V., Yahnin, A.G., 2008. Diurnal behaviour of the ionospheric Alfvén resonator signatures as observed at high latitude observatory Barentsburg (l=15). Annales Geophysicae 26, 2245–2251. doi:10.5194/angeo-26-2245-2008.

38. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research 15, 1929–1958. URL: http://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf.

39. Trakhtengerts, V.Y., Feldstein, A.Y., 1981. Effect of the nonuniform Alfvèn velocity profile on stratification of magnetospheric convection. Geamagn. Aeron 21, 951–953.

40. Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E., 2018. Deep Learning for Computer Vision: A Brief Review. Computational Intelligence and Neuroscience 2018, 7068349. doi:10.1155/2018/7068349.

41. Wilkins, A.H., Strange, A., Duan, Y., Luo, X., 2020. Identifying microseismic events in a mining scenario using a convolutional neural network. Computers & Geosciences 137, 104418. doi:10.1016/j.cageo.2020.104418.

42. Zak, K., 2019. Helper package with multiple U-Net implementations in Keras as well as useful utility tools helpful when working with image semantic segmentation tasks. URL: https://github.com/karolzak/keras-unet.

43. Zhixu, H., 2017. U-net for image segmentation. URL: https://github.com/zhixuhao/unet.